CLAIMS

What is claimed is:

1. A bus isolator for coupling a bus of a first predetermined configuration and signaling protocol to a peripheral via an input/output (I/O) circuit of a second predetermined configuration and signaling protocol, comprising:

a first interface adapted to couple to the bus;

a second interface interface adapted to coupled to the I/O circuit;

a controller coupled to, and receiving signals from, the first and second interfaces;

a memory coupled to the controller for storing the signals received from the first and second interfaces; and

a processing element coupled to the controller and the memory for managing the activity of the isolator,

wherein data is transmitted from the I/O circuit to the bus only in response to a data request from the bus.

2. The isolator of claim 1 wherein data requested by the bus from the I/O circuit is sent via the second interface to the memory and the processor, where the integrity of the requested data is verified and only sent to the bus if it is appropriately formatted, whereby a defective I/O circuit or a defective peripheral coupled to the I/O circuit cannot capture the bus.

3. The isolator of claim 1 wherein the I/O circuit is a commercially available I/O circuit.

4. The isolator of claim 1 wherein the bus is a commercially available bus.

5. The isolator of claim 1 wherein the memory comprises an isolation memory, having a first portion wherein data is received from the first interface and a second portion where data is received from the second interface.

6. The isolator of claim 5 wherein the memory further comprises a program memory and a flash memory.

7. The isolator of claim 1 wherein the processing element comprises a state machine, a simple controller, a microprocessor, a digital signal processor or a combination of first and second lock-step processors.

8. An electronic system, comprising:

a plurality of peripherals;

a plurality of I/O elements, each coupled to one of the peripherals for passing signals to and from the one of the peripherals;

a plurality of isolation elements, each coupled to at least one of the I/O elements for passing signals to and from the at least one I/O element; and

a bus coupled to each isolation element for sending commands and output data to, and receiving requested input data from, the peripherals via the isolation elements and the I/O elements;

wherein each isolation element comprises:

a target interface coupled to the bus for receiving the commands and output data from, and transmitting the requested input data to, the bus;

a master interface coupled to the at least one of the I/O elements for transmitting the commands and output data to, and receiving the requested input data from, the peripheral;

a controller coupled to the target interface and the master interface for controlling transmission of the commands, the inut data, and the ouput data there between;

a memory coupled to the controller for receiving the commands, the output data, and the requested input data; and

a processor coupled to the memory and the controller for managing command and input and output data flow between the bus and the at least one I/O element.

9. The system of claim 8 wherein the memory comprises an isolation memory having one part coupled to the at least one I/O element for receiving data transfer therefrom and another part coupled to the bus for receiving command and data transfer therefrom.

10. The system of claim 9 wherein the memory further comprises program memory containing programs for execution by the processor and flash memory for containing configuration data for the isolation element.

11. The system of claim 8 further comprising a debug port coupled to the processor.

12. A method for coupling a bus and a peripheral via an isolator, comprising:

determining whether there is a message on the bus for the peripheral;

temporarily storing the message in the isolator;

determining whether the message is for output to the peripheral or input from the peripheral; and

if for output to the peripheral, sending the output to the peripheral; and

if for input from the peripheral;

requesting the input from the peripheral;

receiving the input from the peripheral and temporarily storing it in the isolator;

checking the input from the peripheral; and

if valid, transferring the input from the peripheral to the bus; and

if not valid, not transferring the input from the peripheral to the bus.

13. The method of claim 12 further comprising before the first sending step, determining whether the output for the peripheral is in the message or already stored in the isolator.

14. The method of claim 13 wherein if the output for the peripheral is already in the message, further comprising determining whether a bus coupling the peripheral to the isolator is free and if free sending the output to the peripheral and if not free setting an error indicator.

15. The method of claim 13 wherein if the output for the peripheral is already stored in the isolator, further comprising retrieving the stored output and determining whether a bus coupling the peripheral to the isolator is free and if free sending the output to the peripheral and if not free setting an error indicator.

16. The method of claim 12 wherein the checking step comprises determining whether the input from the peripheral is timely and valid and if timely and valid sending the input from the peripheral to the bus and if not timely and valid setting an error indicator.

17. The method of claim 16 further comprising after the sending step, checking to see whether the sending step was successful.

18. The method of claim 12 further comprising after the transferring step, checking to see whether the transferring step was successful.